Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (original): A method for translating from a virtually-addressed bus to a physically-addressed bus, comprising:

presenting a virtual address for a memory line on the virtually-addressed bus;

initiating snoop processing of an intermediary inclusive storage device coupled to the virtually-addressed bus, the intermediary inclusive device capable of storing information related to the memory line from a main memory coupled to the physically-addressed bus;

storing in the intermediary inclusive storage device a pre-fetched memory line including an address tag and data and a pre-fetched status bit, wherein the pre-fetch status bit includes an ON and an OFF indication;

switching the pre-fetch status bit to OFF when the virtual address for the pre-fetched memory line is presented on the virtually addressed bus;

receiving one of a snoop hit and a snoop miss;

if a snoop hit, initiating further snoop processing on local caches coupled to the virtually-addressed bus; and

if a snoop miss, accessing a memory location in the main memory.

Claim 2 (original): The method of claim 1, wherein when a snoop hit occurs, further comprising reading a coherency bit associated with the memory line, and wherein the status of the coherency bit determines a processes for supplying the memory line in accordance with the presented virtual address.

Claim 3 (original): The method of claim 1, wherein memory lines are stored in an intermediary inclusive cache.

Claim 4 (original): The method of claim 1, wherein address tags are stored in a coherency filter.

Claim 5 (currently amended): A method for reducing processing time and bus bandwidth during snoop processing of a multi-processor computer architecture, the architecture comprising higher level caches and intermediary caches, the intermediary caches implemented as coherency filters, the intermediary caches implemented as coherency filters, the method, comprising:

establishing the intermediary caches as inclusive caches, wherein an inclusive intermediary cache includes at least all memory lines of corresponding higher level caches;

presenting a virtual address for a memory line on a virtually-addressed bus; comprising:

entering a tag associated with the memory line into a memory structure of a coherency filter,

entering an identity of a processor that owns the memory line, and

entering a coherency protocol of the memory line;

initiating snoop processing of the intermediary caches;

if receiving a snoop hit, initiating snoop processing on the higher level caches; and

if receiving a snoop miss, accessing main memory.

Claim 6 (original): The method of claim 5, wherein establishing the intermediary caches as inclusive caches comprises making a capacity of the intermediary caches exceed a total capacity of the corresponding higher level caches.

Claim 7 (original): The method of claim 5, wherein establishing the intermediary caches as inclusive caches comprises evicting from any upper level cache a memory line evicted from a corresponding intermediary cache.

Claim 8 (canceled).

Claim 9 (original): A multi-processor computer architecture for reducing processing time and bus bandwidth during snoop processing, comprising:

a plurality of processors;

a plurality of local caches, each local cache corresponding to one of the processors;

one or more virtual busses coupled to the local caches and the processors;

one or more intermediary caches, wherein at least one intermediary cache is coupled to each virtual bus, each intermediary cache comprising:

a memory array, and

means for ensuring the intermediary cache is inclusive of associated local caches; and

a main memory having a plurality of memory lines accessible by the processors.

Claim 10 (original): The architecture of claim 9, wherein the ensuring means comprises a capacity of the intermediary cache equal to or greater than a combined capacity of the associated local caches.

Claim 11 (original): The architecture of claim 9, wherein the ensuring means comprises a protocol that evicts from any local cache, a memory line evicted from a corresponding intermediary cache.

Claim 12 (original): The architecture of claim 9, wherein the memory array is structured to store one or more pre-fetch memory lines, each pre-fetch memory line including: an

address tag;

virtual address bits; and

a pre-fetch status bit, wherein the pre-fetch status bit indicates when a virtual address for the pre-fetch memory line is presented on a virtual bus.

Claim 13 (original): The architecture of claim 9, wherein one of the intermediary caches is a coherency filter.

Claim 14 (original): The architecture of claim 9, wherein one of the intermediary caches is a shared cache.

Claim 15 (original): The architecture of claim 9, further comprising a hierarchy of local caches and intermediary caches.

Claim 16 (original): The architecture of claim 9, further comprising a physical interconnect coupled to each of the intermediary caches.

Claim 17 (original): The architecture of claim 16, wherein the physical interconnect is a cross-bar connection.

Claim 18 (original): The architecture of claim 16, wherein the physical interconnect is a point-to-point link.

Claim 19 (currently amended): A mechanism for translating from a virtual bus to a physical interconnect, comprising:

a main memory storing memory lines;

processors coupled to the main memory and capable of accessing the memory lines; and

means for reducing processing time and bus bandwidth during snoop processing by the processors, wherein the reducing means comprises one or more inclusive cache means coupled to the physical interconnect and to virtual buses, the virtual buses coupled to the processors~~.~~, wherein the reducing means comprises one or more inclusive cache means coupled to the physical interconnect and to virtual buses, the virtual buses coupled to the processors.

Claim 20 (canceled).

Claim 21 (currently amended): The mechanism of claim ~~20~~ 19, wherein the inclusive cache means comprises a capacity of ~~the~~ an intermediary cache equal to or greater than a combined capacity of ~~the~~ associated local caches.

Claim 22 (currently amended): The mechanism of claim ~~20~~ 19, wherein the inclusive cache means comprises a protocol that evicts from any local cache, a memory line evicted from a corresponding intermediary cache.